

TITLE: OVERLAY SCAN LINE PROCESSING

APPLICANT: FONG-SHEK LAM AND KIM A. MEINERTH

CERTIFICATE OF MAILING BY EXPRESS MAILExpress Mail Label No. EL584779051US

I hereby certify that this correspondence is being deposited with the United States Postal Service as Express Mail Post Office to Addressee with sufficient postage on the date indicated below and is addressed to the Assistant Commissioner for Patents, Washington, D.C. 20231.

March 30, 2000

Date of Deposit

Date of Deposit _____
Signature Derek Howard

Signature _____

Derek W. Norwood

Typed or Printed Name of Person Signing Certificate

2025

OVERLAY SCAN LINE PROCESSING

TECHNICAL FIELD

This invention relates to computer display systems, and more particularly to processing overlay scan lines in computer display systems.

BACKGROUND

Conventional computer systems generate pixel maps to represent graphics images. A pixel map is a two dimensional array of pixel values where each pixel value indicates information including color for a corresponding pixel on a monitor or other video display.

Video overlay is the placement of a full-motion video window on the display screen. Video overlay systems can insert into a graphics image a video image such as might be generated by a television tuner, a video camera, VCR, or a video decoder. Video overlay systems commonly include software that generates a pixel map representing the graphics image and provides in the graphics image a video window which is filled with a color key. A separate device such as a video capture card generates the video image.

Current video overlay systems use the horizontal blank time start as an indicator to start processing pixels

for the next overlay scan line. This technique was sufficient with lower resolution monitors that have long horizontal blank times. However, higher resolution monitors and flat panel displays have significantly reduced the amount of horizontal blank time. Thus, higher memory bandwidth is needed to ensure the pixel processing is completed in sufficient time to display the next overlay scan line.

DESCRIPTION OF DRAWINGS

Features and advantages of the invention will become more apparent upon reading the following detailed description and upon reference to the accompanying drawings.

Figure 1 illustrates a computer display including an overlay window according to one embodiment of the present invention.

Figure 2 illustrates a pixel processing engine according to one embodiment of the present invention.

Figure 3 is a flowchart showing the overlay data loading process used by a pixel processing engine according to one embodiment of the present invention.

DETAILED DESCRIPTION

Figure 1 illustrates a computer display 100 including an overlay window 115 according to one embodiment of the present invention. The computer display 100 includes an

overall display 110, an active display 105, the overlay window 115, horizontal active time 120, horizontal blank time 125, a first display line 130, a current overlay display line 135, a second display line 140, a next overlay display line 145, and an overlay display position indicator 150. The active display 105 represents the portion of the computer display 100 visible to the user. The overlay window 115 places full-motion video on the display screen. The overlay window 115 may display, for example, video from a DVD-ROM drive. The overlay window 115 may be positioned at any point in the active display 105.

The overlay window 115 is generated by processing and displaying consecutive overlay display lines. The combination of a plurality of these overlay display lines creates the overlay display window. For simplification purposes, the operation of the overlay display window 115 is described showing a current overlay display line 135 and a next overlay display line 145.

The processing of the overall display 110 is divided into multiple sections, including the horizontal active time 120 and the horizontal blank time 125. The horizontal active time 120 represents the time during which the active display 105 is processed. The active display 105 processes a first line 130 during the horizontal active time 120. When an overlay display window is active, the current overlay display

line 135 is processed during the horizontal active time 120. After the first display line 130 is processed, the overall display 110 waits for a period of time, the horizontal blank time 125, before processing the second display line 140.

5 Previous display systems also waited until the end of the horizontal active time 120 before processing the next overlay display line 145. With more advanced and higher resolution displays, the horizontal blank time 125 is significantly reduced. Thus, higher memory bandwidth is needed to ensure
10 the pixel processing is completed in sufficient time to display the next overlay scan line 145.

To allow additional time to process the next overlay scan line 145 and therefore reduce the need to have increased memory bandwidth, the present invention uses the overlay
15 display position indicator 150. The overlay display position indicator 150 may be located at any location along the current overlay scan line 135. In one embodiment of the invention, the overlay display position indicator 150 is located at approximately the midpoint of the current overlay scan line
20 135. Locating the overlay display position indicator 150 at the midpoint of the current overlay scan line 135 allows the video buffer providing data for the overlay window 115 to be approximately half-empty before beginning the processing for the next overlay scan line 145. By beginning the processing

for the next overlay scan line 145 at the midpoint of displaying the current overlay scan line 135, the next overlay scan line 145 is processed during horizontal active time 120. Of course, when the current overlay scan line 135 is fully displayed, the buffer can begin processing the final portion of the next overlay scan line 145.

Figure 2 illustrates a pixel processing engine 200 according to one embodiment of the present invention. The pixel processing engine includes an input from video memory 205, a vertical zoom (V_{zoom}) 210, a video buffer 215 having a position indicator 220, a horizontal zoom (H_{zoom}), a pixel color conversion and adjustment stage 230, and an output 235 to the display. The pixel processing engine 200 generates the pixel information necessary to display the overlay window 115. The pixel processing engine 200 creates the overlay window 115 by generating a plurality of overlay scan lines.

The pixel processing engine 200 receives video data at an input from the video memory 205. The video data is processed by a V_{zoom} 210. The V_{zoom} 210 is a vertical filter that processes the video data to provide any adjustments in the vertical direction. After processing by the V_{zoom} 210, the video data is sent to a video buffer 215. In one embodiment, the video buffer 215 is a first-in, first-out (FIFO) buffer. The video buffer 215 may include a position indicator 220 showing

the buffer location of the last item of data processed. The video buffer 215 provides storage for the video data until the video data is sent to the display.

After leaving the video buffer 215, the video data is processed by a H_{zoom} 225. The H_{zoom} 225 is a horizontal filter that processes the video to provide any adjustments in the horizontal direction. After processing by the H_{zoom} 225, the video data is sent to the pixel color conversion and adjustment stage 230 for further processing. The pixel color conversion and adjustment stage 230 performs the final processing and adjustment to the video data before being sent to the display. The details of the processing are known to one of skill in the art and will not be discussed herein. After final processing, the video data is provided to the output 235 for transmission to the display.

Figure 3 shows the overlay data loading process 300 used by the pixel processing engine 200 in Figure 2. The process 300 begins at a start state 305. Proceeding to state 310, the process 300 sets the position indicator 220 at a predetermined location in the video buffer 215. In one embodiment, the position indicator 220 is set at approximately the midpoint of the video buffer 215. Of course, the position indicator 220 may be set at any point in the buffer without departing from the spirit of the invention.

Proceeding to state 315, the overlay pixel data is read from the video buffer 215 and provided to the display. The overlay pixel data is used to build the current overlay data line 135 in the overlay window 115. With each bit of pixel data read, the memory location to read from the video buffer 215 is incremented.

Proceeding to state 320, the process 300 determines if the last pixel data was retrieved from the buffer at the indicator location. For example, if the indicator is at the midpoint of the buffer, the current overlay data line 135 in the overlay window 115 will be half-drawn when the buffer memory location reaches the indicator. If the buffer has not reached the indicator, the process 300 proceeds along the NO branch back to state 315. In state 315, the process 300 continues to read data from the buffer to draw the current overlay data line 135. The process 300 remains in this loop until the current overlay data line 135 is drawn to a point where the indicator is reached.

Returning to state 320, if the video buffer has reached the indicator, the process 300 proceeds along the YES branch to state 325. In state 325, the pixel processing engine 200 begins to read data from the video memory for the next overlay data line 140. This loads the video buffer with data for the next overlay data line 145 prior to the completion of

5

1. The first part of the paper discusses the importance of the research and the objectives of the study.